

## Formularios Web con HTML y PHP ver.4

### Índice de contenido

1	Intro.....	2
2	Elemento <form> de un formulario HTML.....	2
3	Dos archivos vamos a crear: el formulario.html y la respuesta.php.....	2
4	Campos de texto.....	3
5	Botones de envío.....	3
5.1	Ejemplo1. Mi color preferido.....	3
5.2	Ejercicio1. Mis favoritos.....	7
6	Casillas de verificación.....	8
6.1	Ejemplo2. Casillas de verificación. ¿has ido a...?.....	8
6.2	Ejercicio2. Ingredientes extras de una Pizza.....	10
7	Botones de opción.....	10
7.1	Ejemplo3. Botones de Opción. Test aritmética.....	11
7.2	Ejemplo3bis. Test aritmética Corregido. IF...ELSE.....	13
7.3	Ejercicio3. Botones de opción. Examen C. Naturales.....	14
8	Ejemplo4. Cuadros de Lista.....	15
8.1	Archivo respuesta con PHP: ejemplo4.php.....	17
8.2	Ejercicio4. Cuadros de lista para tu color favorito.....	18
9	Trabajo Final: Menú del Día.....	18
10	Extra Ejemplo 5. Calculadora para sumar.....	19
11	Ejercicio 5. Financiera de vehículos.....	20
12	OUTRO.....	23

## 1 Intro

Existen dos formas básicas de interactuar con una página web, podemos hacer clic sobre un enlace o enviar datos en un formulario.

Los formularios se usan en la navegación habitual por páginas web (i.e. Cuando entramos en nuestro correo web, iniciamos sesión en *youtube* para subir un vídeo, buscamos una página con *google*, o nos identificamos en *facebook* para acceder a nuestra red social,... siempre estamos enviando datos mediante los campos de un formulario.



Los formularios de una página web se diseñan como el resto de la página, empleando HTML (Lenguaje de Etiquetas de Hipertexto).

## 2 Elemento `<form>` de un formulario HTML

Para crear un formulario basta insertar etiquetas `<form>` de apertura y cierre en tu página WEB. Todos los controles (campos de texto, casillas de verificación, botones de opción o botón de enviar) que añadas entre las etiquetas `<form>` y `</form>` formarán parte del formulario que se envía al servidor WEB.

El elemento `<form>` cuenta con muchos atributos pero basta con utilizar 2 de ellos: **action** y **method**.

**Atributo action:** indica al servidor a qué página (un archivo) debe desplazarse cuando el usuario pulsa el botón de envío del formulario. Es decir, indica qué página recibirá la información del formulario. Por supuesto, esta página debe existir en el servidor Web.

**Atributo method:** controla la forma en que se envía la información al servidor y existen dos métodos, GET y POST. Con GET se envía la información introducida por el usuario adjuntándola a la URL después del signo “?”. Con POST se envían los valores introducidos en el formulario en el cuerpo de la solicitud HTTP.

El código siguiente crea un formulario WEB ... que aún no contiene ningún campo. Cuando añadamos los campos normales en un formulario, el archivo “**respuesta1.php**” recibirá mediante el método POST los datos introducidos en el formulario.

```
<form action="respuesta1.php"method="post">
aquí van los controles del formulario...
...
</form>
```

## 3 Dos archivos vamos a crear: el formulario.html y la respuesta.php

Para todos los formularios que vamos a diseñar crearemos dos archivos.

**Archivo\_1.html.** El primer archivo contiene el formulario donde el usuario introduce en campos la información solicitada y el botón de envío. Es un archivo HTML

**Archivo\_2.php** El segundo archivo recupera los datos del formulario para tratarlos posteriormente (i.e. mostrarlos en la pantalla del ordenador, añadirlos a una base de datos,...). Este archivo Incorpora además de código HTML, otro lenguaje distinto llamado **PHP**, del que hablaremos más adelante.

**Vínculo:** Siempre debemos crear un enlace entre los dos archivos...

Archivo\_1.html → Archivo\_2.php

A continuación vamos a presentar los campos o controles más comunes que podemos usar en un formulario.

#### 4 Campos de texto

Ejemplo de cuadro de texto:

Los campos de texto, también llamados cuadros de texto, son probablemente los controles más conocidos en cualquier formulario. Se crean por medio del elemento

**<input>** y estableciendo dos atributos: **type** igual a **“text”** indica que el campo es tipo texto y **name** igual a **“color\_favorito”** será el nombre de la variable que almacene la dirección de trabajo que solicitamos rellenar.

```
<input type="text" name="color_favorito">
```

En los campos de texto el usuario puede escribir frases de texto completas, lo que les hace muy interesantes para preguntas abiertas (i.e. Dirección donde trabajas).

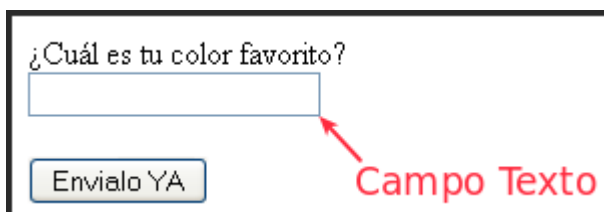
#### 5 Botones de envío

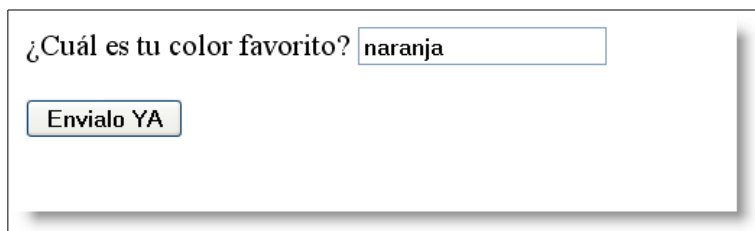
Los botones de envío se usan en todos los formularios. Son necesarios para enviar la información del formulario al archivo.php receptor. Se crean por medio del elemento **<input>** y estableciendo el atributo **type** en **submit**. El texto que aparece sobre el botón se indica mediante el atributo **value=“Envialo Ya”**, y por último, en estos controles no necesitamos definir su atributo nombre (name).

```
<input type="submit" value="Envialo YA">
```

##### 5.1 Ejemplo1. Mi color preferido

Vamos a crear un formulario que te pregunte por tu color preferido y luego lo muestre en pantalla.





¿Cuál es tu color favorito?

Recuerda, necesitamos 2 archivos, uno **HTML** para el formulario y otro **PHP** para mostrar el valor introducido.

Para ello abre tu editor de código favorito (Kompozer, Blue Griffon, Bluefish,...) e introduce el siguiente código:

```
<html>
<head>
  <title>Ejemplo 1. Formulario: Mi Color Favorito</title>
</head>
<body>
  <form method="GET" action="respuesta1.php">
    ¿Cuál es tu color favorito?
    <br>
    <input type="text" name="color">
    <br>
    <input value="Envialo YA" type="submit">
    <br>
  </form>
</body>
</html>
```

Guárdalo en tu carpeta raíz como **ejemplo1.html** y cierra el archivo.

Si introducimos un color en el campo de texto y pulsamos el botón de enviar, obtenemos un ERROR. Aún no funciona el botón de “enviar” ya que no hemos creado la segunda página web, la que recibe los datos introducidos en el formulario.

Como se ha indicado en el formulario `<form action="respuesta1.php">` sabemos que la segunda página se llamará “**respuesta1.php**”.

¿Cómo podemos recuperar la información del color introducido en el formulario en un segundo archivo? Sólo con HTML NO BASTA, ya que NO puede trabajar con variables.

Necesitamos escribir guiones con **PHP**, un lenguaje de programación que trabaja del lado del servidor de páginas web, y que es capaz de manejar las variables que almacenan los datos introducidos en el servidor.

En las próximas líneas vamos a abrir una pequeña ventana al lenguaje PHP, si tienes dudas de mi explicación, hay un tutorial de PHP ameno y muy, muy claro que he encontrado dentro de la sección de informática del sitio web <http://www.rinconastur.net/>

Seguimos con PHP, estos guiones o líneas de código se introducirán dentro del código HTML de la página web, y para separarlas del resto del código irán siempre delimitadas por las etiquetas de

apertura '**<?php**' y de cierre '**?>**'.

El código de nuestra página es el siguiente, introdúcelo en tu editor web y guárdalo como **respuesta1.php**

```
<body>
  Tu color favorito es: <br>
  <?php
    echo $_GET['color'];
  ?>
</body>
</html>
```

¿Que hace nuestro guión PHP?

**<?php** : indica que iniciamos un guión escrito en php.

**echo** : es un comando usado para mostrar en la página web las palabras (cadenas de texto) y/o variables que queramos. Las cadenas de texto deben ir entre comillas (i.e. **echo "hola mundo";**) las variables no lo necesitan (i.e. **echo \$x;**). Si preferimos, para que sea más claro el código usamos paréntesis ( ) para delimitar el contenido que vamos a mostrar en pantalla (i.e. **echo ("Hola mundo");**)

En nuestro ejemplo el comando **echo** mostrará en la página web el contenido de la variable **\$\_GET['color']**,...pero ¿qué es esto? No es difícil, ahora te lo explico...

**\$\_GET['color']** es una variable automáticamente creada por PHP y que almacena el valor del 'color' introducido en el campo de texto. ¿un poco rara? bueno, ya te acostumbrarás... tiene varios símbolos que también te explico:

- **\$** se emplea siempre en PHP para indicar que lo que viene a continuación es una variable.
- **\_GET[' ']** porque el método de envío del formulario elegido es GET, y los corchetes que contendrán el nombre del campo de texto.
- **color** porque es el nombre que le dimos al campo de texto en el formulario( atributo name="color"), y las comillas sencillas ' se usan siempre que trabajamos con cadenas alfanuméricas, es decir, textos.
- **“;”** El punto y coma siempre se usa al final de las instrucciones PHP, es imprescindible ponerla.

*\*Atento: PHP es superquisquilloso con las mayúsculas y minúsculas, para él 'A' y 'a' son distintas: si el nombre de nuestro campo de texto es 'color', no podemos invocarlo en PHP como 'Color', 'coLor',.... ¡ATENCIÓN!.*

Ahora ya tenemos los dos archivos, el primero es el formulario donde nos hacen la pregunta sobre nuestro color preferido con un campo de texto para responderla y la segunda una página llamada **"respuesta1.php"** que mostrará el color elegido. Pero si abres con un navegador el formulario, rellenas el campo de texto con la palabra "naranja" y lo envías pulsando el botón, **NO APARECE TU RESPUESTA "naranja"** en página **"respuesta1.php"**...¿por que?

Tu color favorito es:

**IMPORTANTE:** Bien, si te fijas el archivo “**respuesta1.php**” tiene extensión PHP, estos archivos sólo se pueden interpretar en el servidor de páginas web. Bien pero, ¿Quién tiene un servidor de páginas web en su PC?...

Hoy en día esto es fácil y... gratuito. El servidor web más usado en el mundo es APACHE, tanto en Windows como en Linux. En Windows existen varias *suites* gratuitas que instalan APACHE con un sólo click en tu PC, junto con servidores de bases de datos MySQL, servidor de Correo Mercury y servidores FTP Filezilla.

Una de las *suites* más conocidas en Windows es XAMPP (gratuito), si la instalas, todas las páginas web deben guardarse en la carpeta de Windows “**c:\XAMPP\htdocs\**”, para que nos las sirva el servidor APACHE.

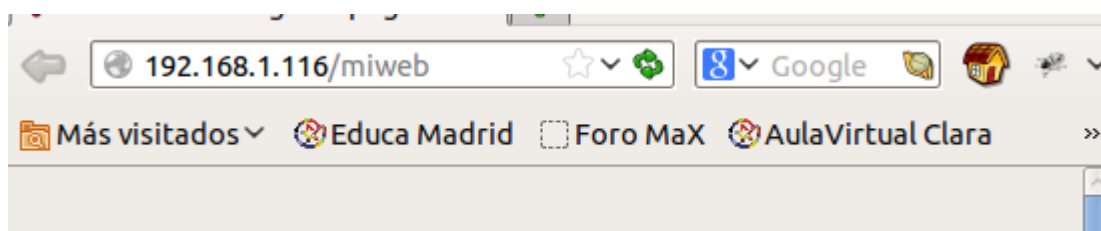
En Linux, y concretamente en la distribución Ubuntu (MAX es Ubuntu), tenemos ya instalado un servidor APACHE, y la carpeta donde debemos guardar las páginas web es “**/var/www/**”. Además, siempre debemos **conceder permisos adecuados** para que nuestra página web sea accesible desde cualquier cliente que la solicite a través de un navegador.

**MÁS IMPORTANTE:** nosotros siempre vamos a copiar todos nuestros archivos de páginas web, sean HTML o PHP al ordenador del profesor que cuenta con un servidor APACHE, a una carpeta compartida que encontrareis en **/miweb/**

APACHE servirá las páginas web que haya en esta carpeta.

**AÚN MÁS IMPORTANTE:** Y si quiero ver de una vez mi formulario en acción...¿Que HAGGOOO?

Bien, abre el navegador web (i.e. Mozilla Firefox, Chrome,...) en tu PC y en el campo de texto de las direcciones URL introduce la dirección IP del servidor donde está el formulario, seguida de **/miweb/** i.e.:



y verás todas las carpetas dentro de **/miweb/**, navega entre ellas para encontrar el archivo del formulario que quieres abrir y haz click sobre él.

Haz click en el **ejemplo1.html**, verás el formulario que has diseñado. Rellena el cuadro de texto y envía el formulario. Verás tu color favorito escrito en la página **respuesta1.php**.

¿Cuál es tu color favorito?

*ejemplo1.html*

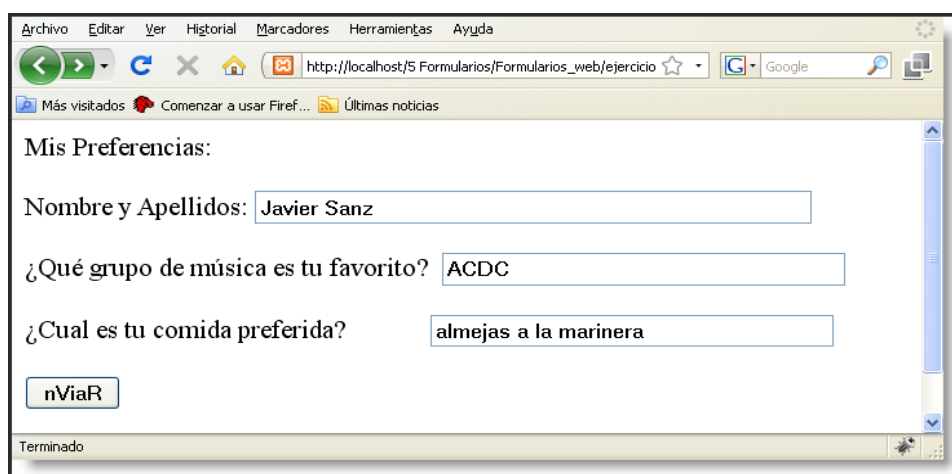
Tu color favorito es:  
naranja

*respuesta1.php*

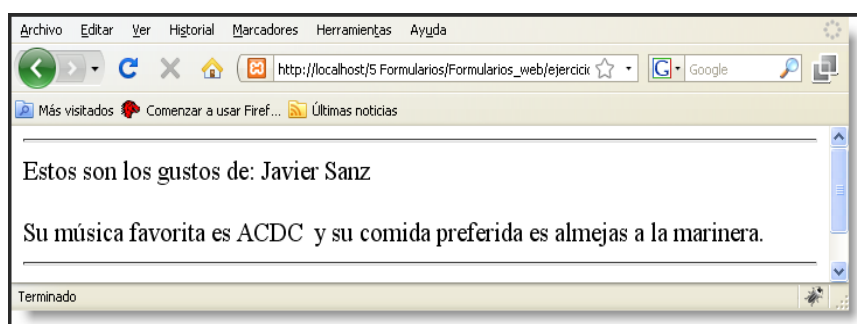
## 5.2 Ejercicio1. Mis favoritos

Te propongo un ejercicio sin mostrarte la solución, así comprobarás si has aprendido bien cómo crear tu primer formulario. Ánimo...

Tarea: Crea un formulario con los campos de texto que ves en la captura y luego crea una sencilla página para mostrar los valores introducidos.



*ejercicio1.html*



*ejercicio1.php*

Si has hecho bien el `ejercicio1` enséñaselo al profesor y continua con el tutorial.

## 6 Casillas de verificación

Es una sola casilla que se puede marcar o nó, según lo que seleccionemos. Para crearla el elemento **<input>** debe tener como atributo **type="checkbox"**, además si queremos que aparezca por defecto marcada la casilla lo indicaremos con **checked**.

Las casillas de verificación son ideales para respuestas del tipo SI/NO. Si necesitamos pasar al archivo.php alguna información más útil que SÍ o NO lo indicaremos en el atributo **value** (i.e. **value="Museo del Prado"**).

### 6.1 Ejemplo2. Casillas de verificación. ¿has ido a...?

Vamos a crear un formulario que pregunte si has estado en algunos museos...

*ejemplo2.html*

Para ello el código HTML es:

```
<body>
  <form method = "GET" action = "ejemplo2.php" >
    Nombre: <input name = "nombre" > y apellidos: <input name = "apellidos" >
    <br> <br>
    ¿alguna vez has ido....
    <br>
    al museo del Prado? <input name = "prado" value = " Museo del Prado" type = "checkbox" >
    <br>
    al Oceanografic de Valencia? <input name = "oceanografic"
    value = "Oceanográfico de Valencia" type = "checkbox" >
    <br>
    <input value = "nViaDo" type = "submit">
    <br>
  </form>
</body>
```



Guarda el archivo como “**ejemplo2.html**” y a continuación vamos a crear la página donde se presentará la información introducida en el formulario.

El archivo “**ejemplo2.php**” recogerá las respuestas marcadas en las casillas y mostrará sus valores (**value**=“Museo del Prado” y **value**=“Oceanográfico de Valencia”). También mostrará el nombre y apellidos separados por un espacio entre ellos con esta línea:

```
<?php echo ($_GET['nombre']." ".$_GET['apellidos']);?>
```

Fíjate bien que después del nombre se añade entre comillas un espacio y luego se añade el apellido. Y fíjate que se usa el punto “.” para **concatenar** (unir) cadenas en la instrucción **echo**.

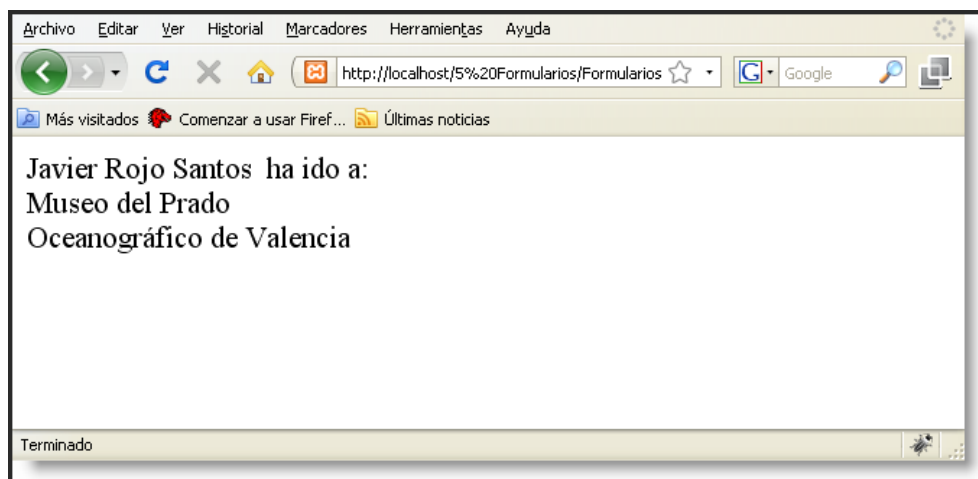
Esto significa que se mostrarán juntos en la página web como una sola línea los tres bloques separados por “.”:

`$_GET['nombre']` seguido de un espacio y luego `$_GET['apellidos']`

El código completo se muestra a continuación, introdúcelo en tu editor web y guárdalo como **ejemplo2.php**.

```
<body>
  <?php echo ($_GET[ 'nombre' ] . " " . $_GET[ 'apellidos' ] ) ; ?>
  ha ido a: <br>
  <?php echo $_GET[ 'prado' ] ; ?>
  <br>
  <?php echo $_GET[ 'oceanografic' ] ; ?>
  <br>
</body>
```

Se debería ver en un navegador como en la captura siguiente:



*ejemplo2.php*

Si todo ha ido bien estás preparado para hacer el ejercicio2...

## 6.2 Ejercicio2. Ingredientes extras de una Pizza

Tarea: Crea un formulario que pregunte si el cliente de una pizzería quiere o no ciertos ingredientes extras. Fíjate en la siguiente captura:

PIZZERÍA "Piemonte"

Mesa nº:

Ha pedido una pizza Especial de la Casa .

Desea alguno de los ingredientes siguientes:

Pimiento: ☒

Queso Gorgonzola: ☐

Rúcola: ☒

Terminado

*ejercicio2.html*

La mesa nº 24 ha pedido los siguientes extras para su pizza:

pimiento

rúcola

Terminado

*ejercicio2.php*

Cuando lo termines y funcione, muéstraselo a tu profesor...

## 7 Botones de opción

Consiste en un círculo que se puede seleccionar o no. Normalmente se usan varios botones juntos y **SÓLO** se puede seleccionar uno de ellos. Es ideal para las respuestas tipo TEST con una sola respuesta posible de entre varias opciones. Los botones de opción se crean con el elemento **<input>** y su atributo **type="radio"**. Es importante que todos los botones "conectados" en el mismo grupo tengan el mismo nombre (atributo **name**). Además, si queremos que uno de los botones aparezca de entrada como marcado se usa el atributo **checked**. La información que se

pasa a la segunda página se puede introducir en el atributo **value="valor que quiera enviar al seleccionar este botón"**, observa el siguiente ejemplo.

### 7.1 Ejemplo3. Botones de Opción. Test aritmética

Vamos a crear un formulario con una pregunta tipo test sobre aritmética. Tendrá 4 respuestas posibles y sólo se podrá elegir una de ellas. Se mostrarán siempre por defecto la primera opción como marcada (necesitaremos usar el atributo **checked**). En la siguiente captura se muestra el formulario:

Y el archivo respuesta

*ejemplo3.html*

*ejemplo3.php*

## DISEÑO DEL FORMULARIO

Introduce este código HTML en tu página Web y guárdalo en tu carpeta raíz como **“ejemplo3.html”**:

```
<body>
<h2>Examen tipo test de aritmética</h2>
  <form method="GET" action="ejemplo3.php">
    Nombre: <input name="nombre"> Apellidos: <input name="apellidos">
    <br><br>
    1. Cuanto es 3+(3x6) ?
    <br>
    a) 17 <input checked="checked" name="pregunta1" value="17" type="radio">
    <br>
```

```
b) 21 <input name="pregunta1" value="21" type="radio">
<br>
c) 16 <input name="pregunta1" value="16" type="radio">
<br>
d) Ninguna de las anteriores respuestas<input name="pregunta1"
value="ninguna de las respuestas mostradas" type="radio">
<br>
<br>
<input value="nViaDo" type="submit"><br>
</form>
</body>
```

### Explicación del código.

Como ya tienes cierto conocimiento en el manejo de formularios iré a lo más importante:

1. Al crear el formulario se muestra el atributo **action** con el nombre del archivo (**ejemplo3.php**) donde se enviará la información del formulario.
2. Fíjate que todos los botones de opción tienen el mismo atributo nombre (**name="pregunta1"**) de esta forma el formulario reconoce que los botones pertenecen al mismo grupo y no permitirá que se seleccione simultáneamente más de uno.
3. En cada botón de opción se define un atributo **value** diferente. Este atributo contiene el texto que se enviará al archivo.php respuesta del formulario cuando se ha elegido ese botón de opción.

**IMPORTANTE:** Si hiciéramos un examen tipo test con varias preguntas sólo habría que nombrar cada grupo de botones como **name="pregunta1"**, **name="pregunta2"**, y así sucesivamente.

Ahora vamos con la página llamada "**ejemplo3.php**" que recoge la información introducida en el formulario. La captura siguiente te la muestra.

El código siguiente consigue recuperar la información del formulario y mostrarla en pantalla. Introdúcelo en tu editor web y guárdalo en tu carpeta raíz como "**ejemplo3.php**".

```
<body>
  El alumno <?php echo $_GET['nombre'] . " " . $_GET['apellidos'] ; ?>
  ha respondido que 3+(3x6) es:
  <br>
  <?php echo $_GET['aritmetica'] ; ?>
  <br>
  <br>
</body>
```

No hay mucho que explicar, el primer guión PHP muestra en pantalla mediante el comando **echo** el nombre y apellidos, como ya hemos visto en otros ejemplos. Y la línea `<?php echo $_GET['aritmetica'];?>` muestra el valor del botón (atributo **value="17"**) que se ha seleccionado.

Después de hacer este ejemplo sobre botones de opción para hacer preguntas tipo test, uno se hace siempre la misma pregunta. ¿Podríamos saber si la respuesta seleccionada es correcta? Por supuesto que sí, pero para ello necesitamos manejar otros comandos de PHP, las sentencias

## IF ELSE

Son sencillas de emplear, en el siguiente ejemplo lo abordamos.

### 7.2 Ejemplo3bis. Test aritmética Corregido. IF...ELSE

Añade el código PHP a tu página **ejemplo3.php** y guárdalo como **“ejemplo3bis.php”**.

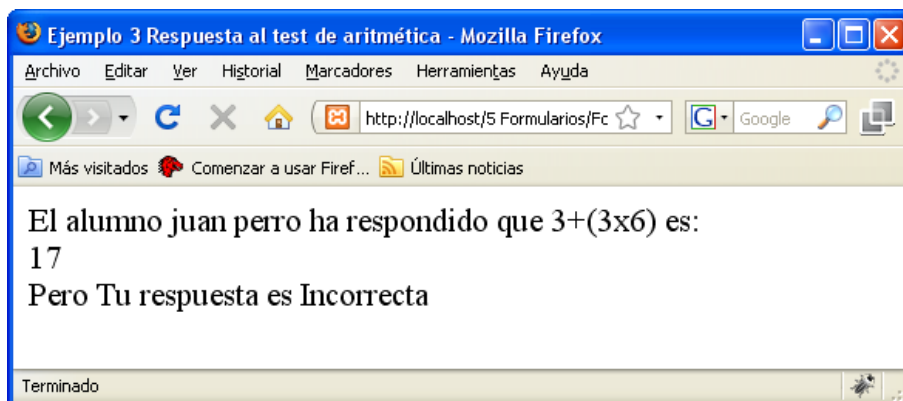
```
<body>...
  <body>
    El alumno ..... ha respondido que .....es: <br/>
    <?php echo $_GET['aritmetica'];
        if ($_GET['aritmetica']==21) {echo ("Tu respuesta es correcta") ; }
        else {echo ("<br/> Pero Tu respuesta es Incorrecta") ; }
    ?>
  </body>
```

La sentencia IF...ELSE (SI...SI NO) funciona así:

SI (se cumple esta condición) {hago esto;} SI NO {hago lo otro;}

En el código anterior:

Si la variable `$_GET['aritmetica']` es igual a 21 muestra en pantalla “Tu respuesta es correcta” si no muestra “Pero Tu respuesta es Incorrecta”.



*ejemplo3bis.php*

Si lo has entendido, ¡¡¡ ENHORABUENA, estás en el camino de ser un programador...!!!

### 7.3 Ejercicio3. Botones de opción. Examen C. Naturales

Tarea: Crea un examen de Ciencias Naturales como se muestra en el siguiente formulario y guárdalo como **ejercicio3.html**

Ejercicio 3 Test de C. Naturales - Mozilla Firefox

Archivo Editar Ver Historial Marcadores Herramientas Ayuda

http://localhost/5 Formularios/Formulario: Google

Más visitados Comenzar a usar Firef... Últimas noticias

## Examen tipo test de C. Naturales

Nombre:  Apellidos:

1. ¿ Cuantos planetas hay en nuestro sistema solar?

- a) 7 planetas ☒
- b) 8 planetas ☐
- c) 9 planetas ☐
- d) Ninguna de las anteriores ☐

2. Los quelonios son:

- a) Ranas ☐
- b) Culebras ☒
- c) Tortugas ☐
- d) Ninguna de las anteriores ☐

Terminado

*ejercicio3.html*

El archivo que recoge las respuestas se llamará **ejercicio3.php**, a continuación te muestro una captura de al lado:

Ejercicio 3 Respuesta al test de Ciencias Naturales - Mozilla Firefox

Archivo Editar Ver Historial Marcadores Herramientas Ayuda

http://localhost/5 Forml... Google

Más visitados Comenzar a usar Firef... Últimas noticias

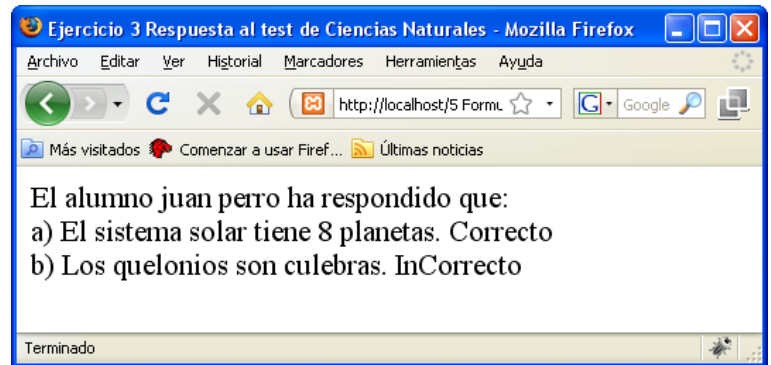
El alumno juan perro ha respondido que:

- a) El sistema solar tiene 7 planetas.
- b) Los quelonios son culebras.

Terminado

*ejercicio3.php*

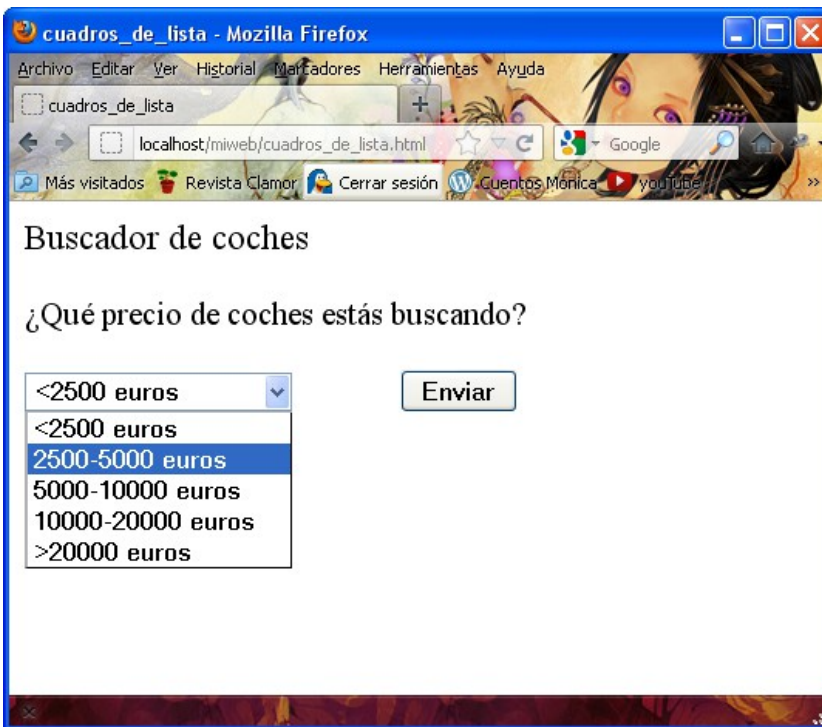
Intenta que este archivo **ejercicio3.php** muestre si las respuestas son correctas o no, como se muestra en la captura siguiente.



Guárdalo como **ejercicio3bis.php**

*ejercicio3bis.php*

## 8 Ejemplo4. Cuadros de Lista



*ejemplo4.html*

lista en un formulario....

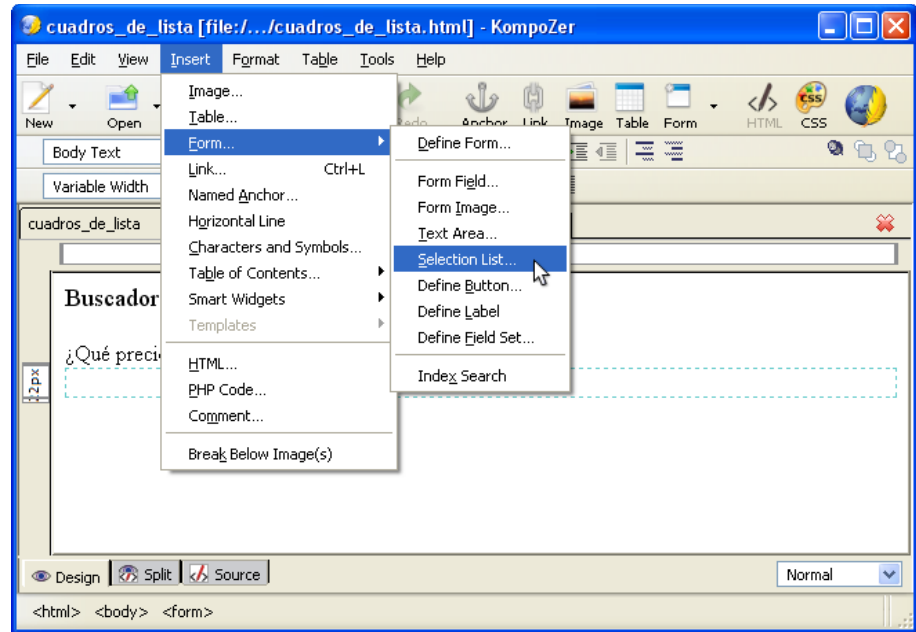
También llamados **cuadros de lista desplegables** son controles que muestran en una lista varios elementos. Funcionan básicamente igual que los botones de opción agrupados, donde sólo se puede escoger uno de ellos.

Creamos una archivo llamado **ejemplo4.html** y definimos en él un formulario con sus atributos **action** y **method** bien definidos como hacemos siempre.

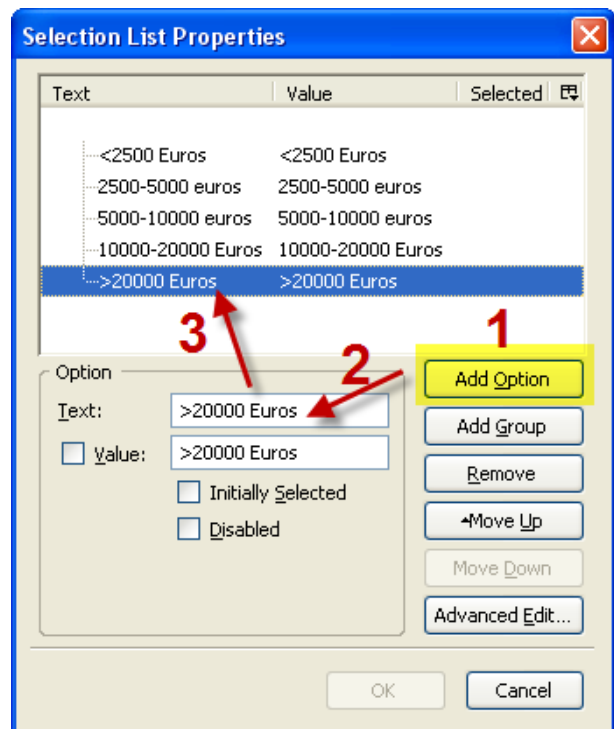
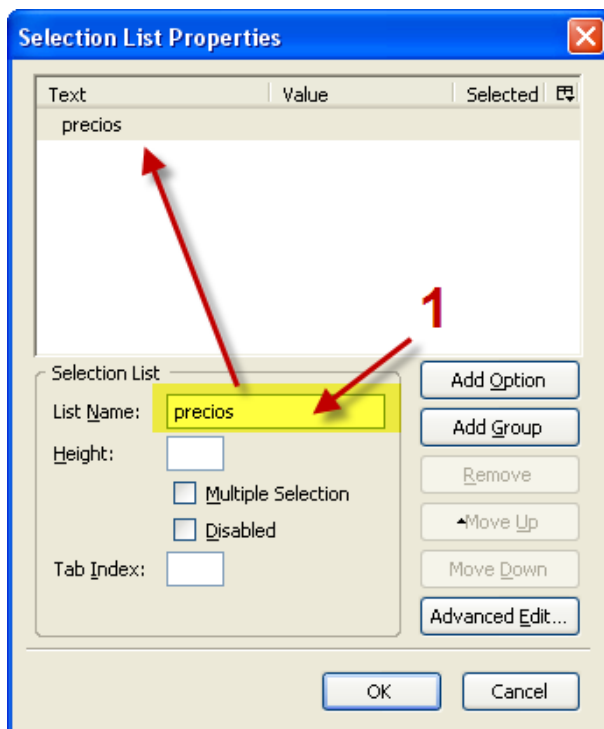
El cuadro de lista es un control un tanto especial, ya que se define con una etiqueta **<select>** y antes de cerrarla definimos cada elemento de la lista con la etiqueta **<option>**. Con Kompozer podemos usar los menús para insertar automáticamente los cuadros de



Después de haber definido un formulario, insertamos un Cuadro de lista (Selection List si el menú está en inglés), como se muestra en la captura:



Se abrirá un cuadro de diálogo con las propiedades del cuadro de lista. Lo primero que tenemos que hacer es definir el nombre del cuadro de lista, como se muestra en la captura:



A continuación insertamos el primer elemento de la lista desplegable haciendo click en el botón "Insertar Opción" y tecleando el nombre del primer elemento que aparecerá en mi lista. Repetimos el proceso para insertar los 5 elementos de esta lista. Para definir los elementos de la lista sigue los pasos mostrados en la captura de arriba.

Si todo ha salido bien tendremos un código html como el siguiente:

```
<html>
<head> <title>cuadros_de_lista</title>
```



```
</head>
<body>
<big>Buscador de coches</big><br>
<br>
¿Qué precio de coches estás buscando?<br>
<br>
<form method="get" action="ejemplo5.php">
  <select name="precios">
    <option><2500 euros</option>
    <option>2500-5000 euros</option>
    <option>5000-10000 euros</option>
    <option>10000-20000 euros</option>
    <option>>20000 euros</option>
  </select>
  <input name="enviar" value="Enviar" type="submit"><br>
</form>
</body>
</html>
```

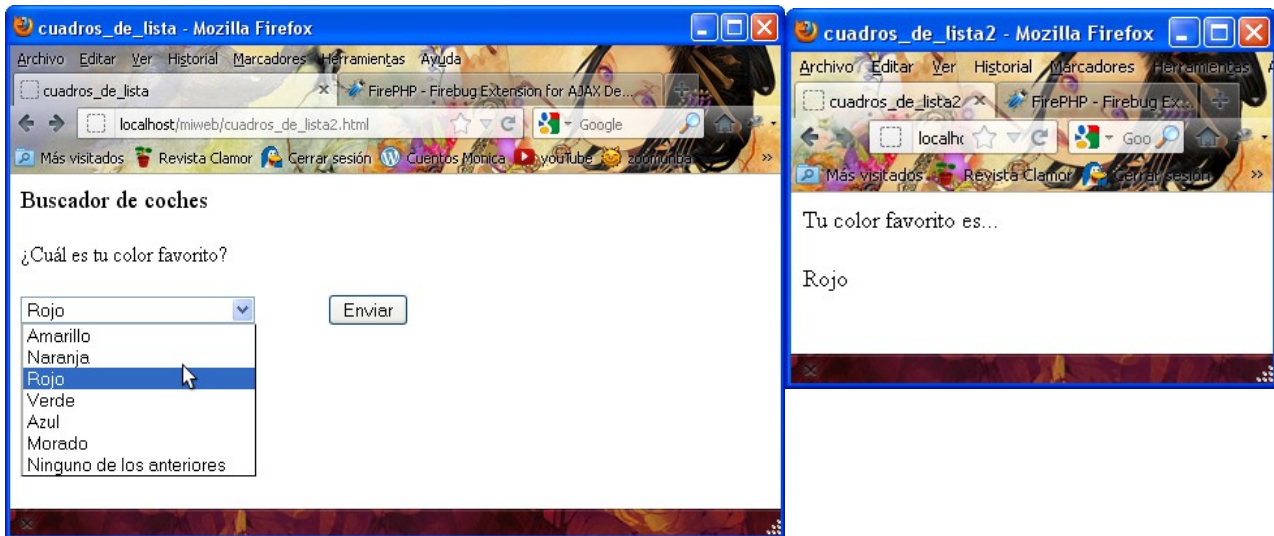
### 8.1 Archivo respuesta con PHP: ejemplo4.php

Creamos un archivo con esta nombre y extensión (PHP) con algún editor de programación (i.e. Bluefish, blue griffon, Kompozer,...). Insertamos un texto inicial explicativo y luego insertamos el código PHP para mostrar en pantalla. No voy a resolver esta parte pues ya lo hemos hecho extensamente en anteriores ejemplos. Te dejo la captura final del archivo respuesta para que lo hagas igual.



## 8.2 Ejercicio4. Cuadros de lista para tu color favorito.

Crea dos archivos ejercicio4.html y ejercicio4.php con el formulario que ves a continuación:



## 9 Trabajo Final: Menú del Día

Crea un formulario que muestre el menú del día en un restaurante, tendrás que pedir los siguientes datos:

1. Número de la mesa (cuadro de texto)
2. Nombre y apellidos del camarero que atiende la mesa (cuadro de texto)
3. Tres primeros platos a elegir uno de ellos (botones radiales)
4. Tres segundos platos a elegir uno de ellos (botones radiales)
5. Tres bebidas a elegir una de ellas. (lista desplegable)
6. Extras como pan, mantequilla, puro pudiendo escoger los que quieras. (con botones checkbox)

Guarda el archivo como **trabajo\_final.html** en tu carpeta raíz

Crea otro archivo para mostrar el menú escogido y guárdalo como “**trabajo\_final.php**” en la misma carpeta raíz.

## 10 Extra Ejemplo 5. Calculadora para sumar

Vamos a crear un formulario que nos pida introducir dos números, para mostrarnos luego en el archivo respuesta su suma...

El archivo del formulario se llamará **ejemplo5.html** y en él los valores a sumar se introducen por medio de dos cuadros de texto:

- Campo de texto X: **name="x"**
- Campo de texto Y: **name="y"**

En la captura se muestra un botón nuevo que vale para **LIMPIAR** los datos que se han introducido en los cuadros de texto. Se hace con el **BOTÓN RESET**.

Este botón de **RESET** lo puedes encontrar en la misma lista de insertar campos de un formulario justo debajo del botón de envío....

Te muestro el código **HTML** del **ejercicio5.html**

```
<html>
<head>
<title>Calculadora</title>
</head>
<body>
    <form action="ejercicio5.php" method="get">
        Esta calculadora suma dos números:<br /><br />
```

```
Intruduce los valores:<br />
X : <input type="text" name="x" /> <br />
Y : <input type="text" name="y" /> <br /><br />
<input type="submit" value="Calcular" />
<input type="reset" value="Borrar" />
</form>
</body>
</html>
```

En corazón de la calculadora está en el archivo de PHP que llamaremos **ejercicio5.php** Te muestro una captura del código para que lo hagas...

```
<html>
<head>
<title>Resultado de la suma</title>
</head>
<body>
  <?php
    $z=$_GET['x']+$_GET['y'];
    echo "La suma de " . $_GET['x'] . " y " . $_GET['y'] . " es: <br />". $z ;
  ?>
</body>
</html>
```

**Explicación:** defino una nueva variable \$z igual a la suma de X e Y, luego muestro en pantalla X, Y y Z.... y se acabó el misterio.

**!!! NO COPIES EL CÓDIGO DE ARRIBA, no aprenderás nada y el día del examen NO PODRÁS HACERLO!!!!**

## 11 Ejercicio 5. Financiera de vehículos.

Vas a crear el formulario de una empresa financiera que presta dinero para que los clientes puedan comprar coches. El formulario de entrada se llamará **ejercicio5.html**

The screenshot shows a Mozilla Firefox browser window with the title 'McAuto financiación'. The address bar shows 'localhost/miweb/ejercicio6.html'. The page content includes a form for car financing with the following fields and options:

- Nombre:** Antonio
- Apellidos:** Júcar
- Edad:** 20
- Dirección:** C/ Olvido, nº 5, 28056 Madrid
- Precio del coche (€):** 15600
- Entrada dispuesto a pagar:** A dropdown menu with options: 5.000 euros, 1.000 euros, 5.000 euros (selected), and 10.000 euros.
- ¿En cuantas mensualidades desea devolver el préstamo?**
  - ☒ 12 meses al 8.0 %
  - ☐ 36 meses al 16.0 %
- Buttons:** 'enviar solicitud de financiación' and 'RESET'.

### Datos para diseñar los controles:

Campo de texto Nombre: **name="nombre"**

Campo de texto Apellidos: **name="apellidos"**

Campo de texto Edad: **name="edad"**

Campo de texto Precio del coche: **name="precio"**

Área de Texto Dirección: **name="direccion"**

Cuadro de Lista Desplegable:

nombre de la lista: **name= "entrada"**

Los valores pasados para cada caso son:

10,000 euros → **value= 1000**

5,000 euros → **value= 5000**



10,000 euros → **value= 10000**

Datos de los **botones de opción agrupados**:

**name="mensualidades"**

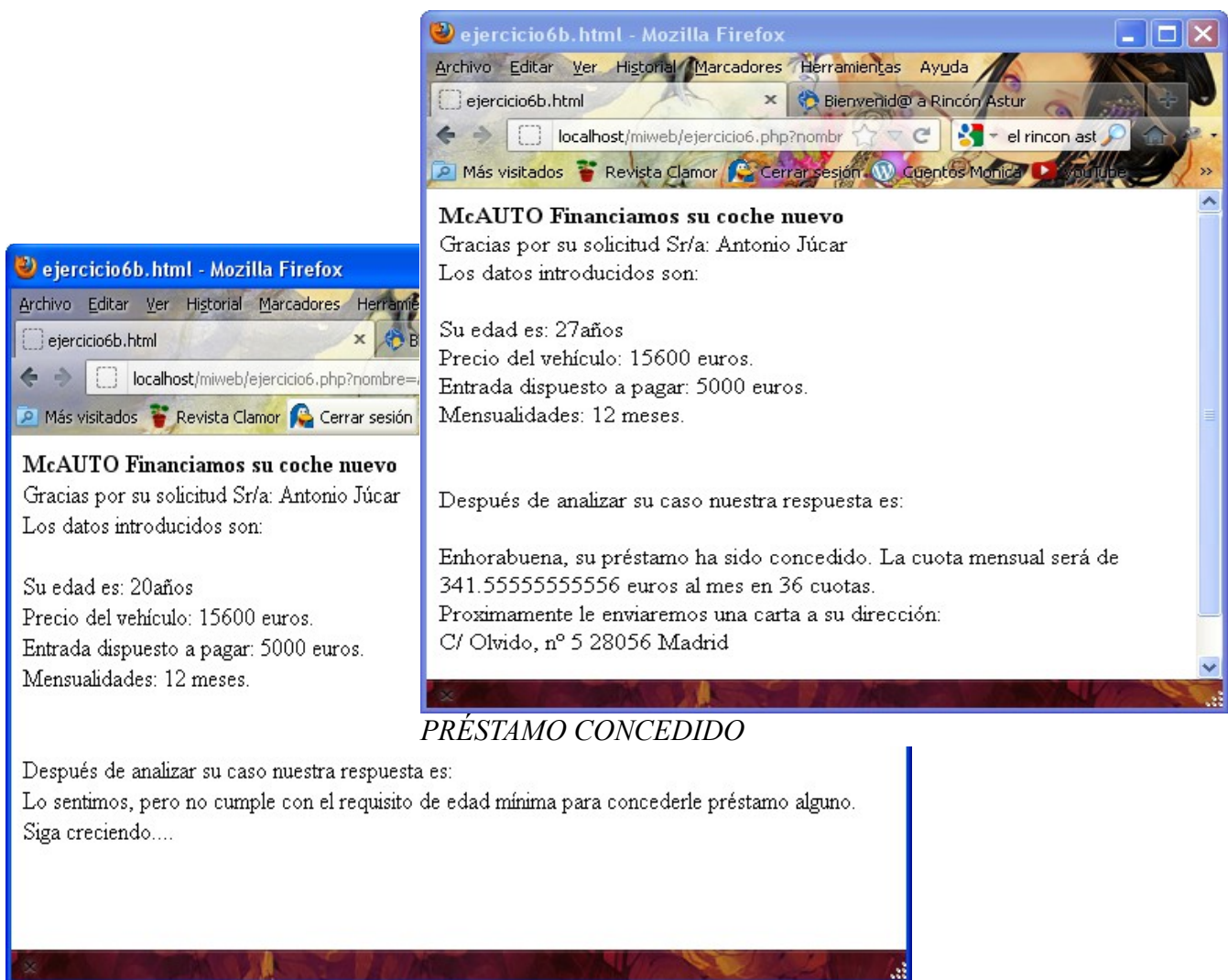
Valores pasados por los botones

12 meses al 8% → **value = 12**

36 meses al 16% → **value = 36**

## Archivo de respuesta al formulario, **ejercicio5.php**

Te muestro dos capturas, una concediendo el préstamo y otra negándolo.



## Código PHP:

- **Recogida de Datos:** Primero tenemos que recuperar los datos introducidos por el solicitante, nombre, edad, precio del vehículo, entrada que pagaremos, y número de mensualidades. Y todas los mostramos tal y como aparecen en cualquiera de los dos páginas respuesta de arriba.
- **Análisis de los datos:**
  - No se le concederá préstamo en ningún caso SI su edad no supera los 25 años.
  - Si cumple con el criterio de la edad...
    - ♦ La cuota a pagar cada mes será igual al precio del vehículo menos lo que pagamos de entrada, todo dividido entre las mensualidades y aplicado el interés correspondiente (8% en el caso de 12 mensualidades o 16% en el caso de 36 mensualidades)  
$$\text{cuota} = \text{interés} * (\text{precio} - \text{entrada}) / \text{mensualidades}$$
    - ♦ Los intereses son fáciles de aplicar, por ejemplo para un interés del 8% se harían los siguientes cálculo:  $\text{Cuota} = 1,08 * (\text{precio} - \text{entrada}) / \text{mensualidades}$

**Pista 1:** cuando una variable (i.e. el interés) puede tomar uno de entre dos valores según la opción escogida podemos usar la estructura IF.... para asignarle los valores.

## 12 OUTRO

Evidentemente, cualquiera que haya visto un formulario sabe que nos hemos saltado algunos controles muy usados, campos para contraseñas, envío de archivos, botones varios.... pero NO es nuestro empeño verlos todos. Hay cientos de ejemplos en páginas web para aprender, ahora la pelota está en tu TEJADO si quieres seguir...

También es evidente que con la información que se introduce en los formularios se puede hacer muchas más cosas que simplemente presentarlas en pantalla. Te he mostrado por una rendija el uso de PHP como corrector de exámenes, pero si piensas un poco, reconocerás la misma utilidad cuando te conectas para leer tu correo, ya que sólo se te permite acceder a tu cuenta de correo cuando introduces el par usuario/contraseña correctos. Esto es una aplicación más de PHP, los *login* o autenticaciones de usuario. Y así hay muchas, muchas mas aplicaciones. Como he dicho antes, si te gusta verás que es mucho más fácil de lo que te ha parecido hasta ahora, y es que los comienzos siempre son lo más difícil.